

4/3 Rectangle Tiling lower bound*

GRZEGORZ GŁUCH, KRZYSZTOF LORYŚ

University of Wrocław

Abstract

The problem that we consider is the following: given an $n \times n$ array A of positive numbers, find a tiling using at most p rectangles (which means that each array element must be covered by some rectangle and no two rectangles must overlap) that minimizes the maximum weight of any rectangle (the weight of a rectangle is the sum of elements which are covered by it). We prove that it is NP-hard to approximate this problem to within a factor of $1\frac{1}{3}$ (the previous best result was $1\frac{1}{4}$).

I. INTRODUCTION

RTILE problem. Given an $n \times n$ array A of positive numbers, find a tiling using at most p rectangles (that is rectangles which cover A without overlap) that minimizes the maximum weight of any rectangle (the weight of a rectangle is the sum of elements which are covered by it). A tile is any rectangular subarray of A .

Previous work. The problem was considered by Khanna, Muthukrishnan and Paterson in [KMP98] where a $2\frac{1}{2}$ upper bound was given. Next it was improved, firstly to $2\frac{1}{3}$, independently by Sharp in [S99] and by Loryś, Paluch in [LP00], and later to $2\frac{1}{5}$ by Berman, DasGupta, Muthukrishnan and Ramaswami in [BDMR01]. The best known result for the upper bound is $2\frac{1}{8}$ and was presented by Paluch in [P04]. The only known lower bound, equal to $1\frac{1}{4}$, was given in [KMP98].

In this paper we obtain a $1\frac{1}{3}$ lower bound for the RTILE problem. As the core of the proof we use a modified construction of the one used in [KMP98].

The main result of the paper is the following theorem.

Theorem 1. *The RTILE problem is NP-hard,*

even in the case where the element weights are integers in the range $[1, 3]$. Furthermore it is NP-hard to determine the optimal value to within a factor of $1\frac{1}{3}$.

II. PROOF OF THE THEOREM

In the proof we reduce PLANAR-3SAT into RTILE. It was shown in [L82] that PLANAR-3SAT is NP-complete. An instance of PLANAR-3SAT problem is a 3CNF formula F with an extra property that the following graph G_F is planar. The bipartite graph G_F has variables and clauses as its two vertex sets. An edge (x, c) exists in G_F if and only if x or $\neg x$ occurs in clause c .

For any formula F , an instance of PLANAR-3SAT problem, we will construct an array A_F of integers in the range $[1, 3]$ and an integer p_F , which will give us an instance $(A_F, p_F, 3)$ of RTILE problem. We will construct this instance in such a manner that F is satisfiable if and only if A_F can be tiled by p_F tiles of weight at most 3.

I. Intuitions

Given F which is an instance of PLANAR-3SAT we want to construct an array A_F . For every

*This work was supported by the Polish National Science Centre grant DEC-2012/06/M/ST6/00459.

variable in F we want to create a closed loop of orthogonally adjacent fields so that every two loops are disjoint (they don't even touch each other). We also want to place these loops on the A_F in such a way that for every clause C in F , the three loops corresponding to variables occurring in C meet in a special *clause gadget* (Fig. 1).

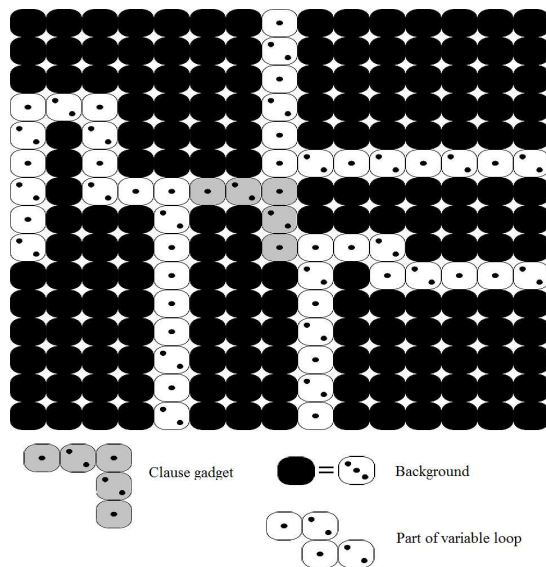


Figure 1: *Clause gadget and variable loops*

Then every loop is filled with numbers in such a way that it is possible to cover the loop in only two different ways (Fig. 2, Fig. 3) (we will call them modes) which corresponds to a valuation of a variable.

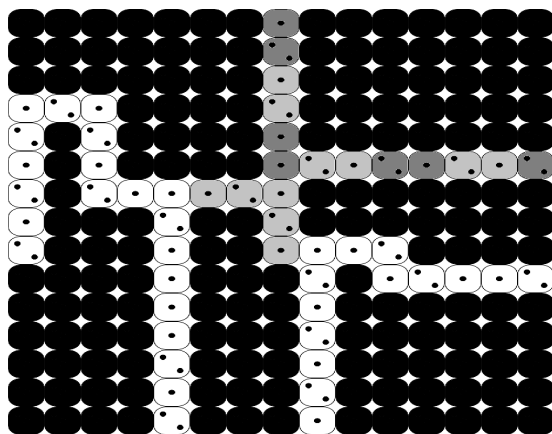


Figure 2: *Loop covering 1*

As we can see in most cases one tile covers only two consecutive elements of a variable loop. What is more if a loop is covered in one mode then part of the clause gadget (namely one of its 1's) can be joined to a tile from the loop (as in the Fig 2.) but it is impossible when a loop is in the other mode (as in the Fig 3.). Situation in the Fig 2. corresponds to a fact that under this valuation that variable causes the clause to evaluate to true. Later we will say that a given covering mode of a loop is in a good phase in proximity of a gadget clause if it is possible to join a 1 from this gadget to a tile from the loop. Finally it is necessary to use three tiles to cover clause gadget if none of its 1's is joined with a variable loop and only two tiles when one or more of its 1's is joined with a loop.

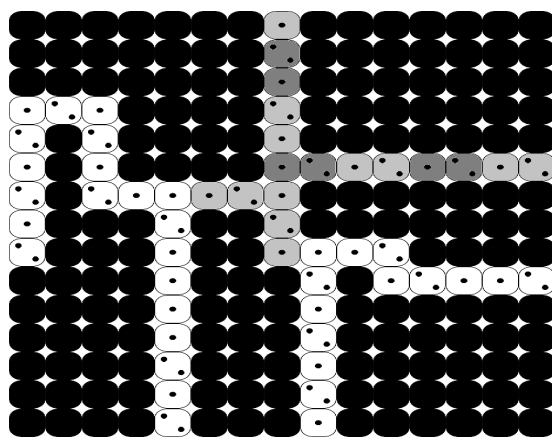


Figure 3: *Loop covering 2*

II. Construction

At first it is possible to construct a topology described above because of the Schnyder theorem [S90] which states that every graph with $n \geq 3$ vertices has a straight line embedding on the $n - 2$ by $n - 2$ grid so that each vertex lies on a grid point.

Next we need to show how to choose values of A_F elements. The background is simply filled in with 3's while each clause gadget is filled as can be seen at Fig. 1. Next the exact filling of variable loops proceed as follows. Each variable loop is filled independently, so let's consider one of them and call it Z . At first we fill neigh-

bourhoods of clause gadgets of Z loop with a sequence 2112 in such a manner that two 1's from 2112 sequence and a 1 from clause gadget form a 3×1 rectangle (Fig. 1). Next we traverse the loop clockwise filling it with alternating 1's and 2's. When filling the loop we may sometimes need to use a sequence 2112 to avoid two consecutive 2's in the loop. What is more we need to provide that being in a good phase near one gadget implies being in a good (or bad) phase in proximity of another gadget depending on whether the corresponding variable occurs in a clause with a negation or not. To assure that property we use (or not) a sequence 211112 (called *phase changer*) between consecutive clause gadgets.

One last thing is the choice of p_F . Denoting the number of variables as n , the number of clauses as k , the number of elements with value 3 as t , clause gadget as C_j , loop corresponding to a variable x_i as Z_i , length of a variable loop Z_i as $|Z_i|$ and the number of *phase changers* in Z_i as $P(Z_i)$ we define $p_F = \frac{1}{2}(\sum_{i=1}^n (|Z_i| - P(Z_i))) + 2k + t$.

III. Correctness of the Construction

Fact 1. *Every closed loop on a rectangle grid has even length.*

Proof. We can treat the array A_F as a bipartite graph G where vertices correspond to elements of A_F and vertices are connected when corresponding elements are orthogonally adjacent. We can see then that a loop corresponds to a cycle in G and because of that is of an even length. \square

Fact 2. *For every variable loop Z_i the number of phase changers $P(Z_i)$ is even.*

Proof. Firstly let us notice that a decision of placing each phase changer was made solely based on the length of a variable loop between two consecutive clause gadgets and negations of the corresponding variable in two appropriate clauses. A phase changer was placed in a segment when the length of that segment was even and negations were different or when the length was odd and negations were the same.

Let $Diff$ be the number of segments between two consecutive clause gadgets where the corresponding variable appeared with different negations and Odd be the number of segments of odd lengths. Both values: Odd and $Diff$ are even, which follows from the Fact 1. and the fact that every variable loop is a cycle. Finally the number of phase changers placed is equal to $Odd - Diff + 2x$ (where x is the number of segments of even lengths with different negations at the ends). \square

Lemma 1. *p_F is an integer.*

Proof. It follows easily from Fact 1 and Fact 2. \square

Theorem 2. *F is satisfiable if and only if A_F can be tiled by p_F tiles of weight at most 3.*

Proof. Implication to the right is obvious, we translate valuations of variables into tiling modes of variable loops getting a valid tiling.

Now we will prove an implication to the left.

Lemma 2. *Every variable loop Z_i is covered by at least $\frac{1}{2}(|Z_i| - P(Z_i))$ tiles.*

Proof. We prove it by contradiction. Let a_i be the number of tiles which covers exactly i elements of a variable loop Z_i . Then we have a series of (in)equalities:

- $a_1 + a_2 + a_3 < \frac{1}{2}(|Z_i| - P(Z_i))$
- $a_3 \leq P(Z_i)$
- $a_1 + 2 * a_2 + 3 * a_3 = |Z_i|$

After solving this set of (in)equalities we get a contradiction. \square

Fact 3. *Every C_j is covered by at least 2 tiles disjoint with those covering variable loops.*

Proof. That is an obvious fact already mentioned in 'Intuitions' section. \square

Lemma 2, Fact 3 and construction of p_F give us that we need to use exactly $\frac{1}{2}(|Z_i| - P(Z_i))$ tiles for covering a variable loop Z_i . That means that every Z_i is covered by $P(Z_i)$ tiles covering three elements of a loop and $\frac{1}{2}(|Z_i| - 3 * P(Z_i))$ tiles covering two loop elements. Then we see that choosing one tile of a loop determines the whole tiling of that loop, or in other words

it determines a covering mode. It comes from the fact that the only places where one tile can cover more than two elements of a loop are in *phase changers*.

Now we want to translate a tiling of a variable loop Z_i using $\frac{1}{2}(|Z_i| - P(Z_i))$ tiles into a valuation of a corresponding variable x_i . If there is no clause gadget such that one of its 1's is joined to the loop Z_i then the valuation of x_i can be chosen arbitrary. So let's say a 1 from a clause gadget C_j is attached to the loop Z_i . Then we choose the valuation of x_i , so that the literal in which x_i occurs in C_j evaluates to true.

Lemma 3. *Choice of valuation of x_i is consistent among Z_i . It means that whenever some 1 from a clause gadget $C_{j'}$ is attached to the loop Z_i then x_i appears in $C_{j'}$ with the same "negation" as in C_j .*

Proof. The 1 attached to Z_i near C_j forces the loop to be covered as in the Fig 2. in proximity of C_j , which in turn forces the covering mode of Z_i . Now let's consider next clause gadget after C_j in clockwise order (namely C_{j+1}). We assured during the construction that being in a good phase near one clause gadget forces being in a good/bad phase in proximity of the next clause gadget. That means that if the variable x_i occurs in the C_{j+1} with the same "negation" as in C_j (clauses look both like $\dots \vee x_i \vee \dots$ or like $\dots \vee \neg x_i \vee \dots$) then Z_i is also in a good phase near C_{j+1} . If x_i occurs with different "negations" in C_j and C_{j+1} then Z_i is in a bad phase near C_{j+1} . By easy inductive argument we get that it is only possible for some other 1 from some other clause gadget C_l to be attached to the Z_i loop when x_i occurs in C_l with the same negation as in C_i . \square

In this manner we constructed a valuation

that satisfies F, what concludes the implication to the left. \square

IV. Conclusions

As we can see the problem of finding RTILE lower bound is far from being solved. Even the existing techniques when applied in slightly more rigorous and careful manner can lead to better results. We also hope that our result will stimulate further research in this area.

REFERENCES

- [KMP98] S. Khanna, S. Muthukrishnan, M. Paterson. On Approximating Rectangle Tiling and Packing. *SODA 1998*.
- [P04] K. Paluch. A 17/8-Approximation Algorithm for Rectangle Tiling. *ICALP 2004* 1054-1065.
- [S99] J. Sharp. Tiling Multi-dimensional Arrays. *Proc. 12th FCT, Springer, 1999, LNCS 1684, 500-511*.
- [LP00] K. Lorys, K. Paluch. Rectangle Tiling. *Proc. 3rd APPROX, Springer, 2000, LNCS 1923, 206-213*.
- [BDMR01] P. Berman, B. DasGupta, S. Muthukrishnan, S. Ramaswami. Efficient Approximation Algorithms for Tiling and Packing Problems with Rectangles. *J. Algorithms* 41(2): 443-470 (2001).
- [L82] D. Lichtenstein. Planar formulae and their uses. *SIAM J. Computing* 11, 329-343, 1982.
- [S90] W. Schnyder. Embedding Planar Graphs on the Grid. *SODA 1990, 138-148*.